

2

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER #26	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Language Design, Computers and Brains		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Henry Kucera		8. CONTRACT OR GRANT NUMBER(s) N00014-81-K-0136
9. PERFORMING ORGANIZATION NAME AND ADDRESS Center for Neural Science Brown University Providence, Rhode Island 02912		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR-201-484
CONTROLLING OFFICE NAME AND ADDRESS Personnel and Training Research Program Office of Naval Research, Code 442PT Arlington, Virginia, 22217		12. REPORT DATE March 29, 1985
MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		13. NUMBER OF PAGES 21 pages
		14. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited. Publication in part or in whole is permitted for any purpose of the United States Government.

DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

DTIC
ELECTE

APR 26 1985

B

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Natural Language, Polysemy,
Information Theory, Markedness,
Formal Grammars, Language Learning, Language Translation,
Redundancy, Abduction as a Form of Inference,
Ambiguity in Language, Sequential vs. Parallel Processing.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The processing of natural languages by computers and the programming of computers to simulate some aspects of language learning offer insights into those aspects of language design which allow successful processing to take place as well as those properties of natural languages which make their computer-based analysis difficult. In essence, these factors represent the differences between the two "information processors" involved in all such endeavors, the biological one, the human brain, of which natural languages are an evolutionary product, and the digital computer. The properties of language which allow

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

85 4 1 078

AD-A152 819

DTIC FILE COPY

computer processing of language data are those of structure, specifiable by a grammar which captures systematically the redundancy in language organization. The other aspect of natural language is its striving towards a reasonable degree of efficiency, a principle which operates throughout some aspects of language change.

The difficulties encountered in language processing by computers center around ambiguity, which can be either structural or semantic. The theory of markedness is described which posits a systematic use of semantic ambiguity for the sake of efficient communication. Finally, a project conducted at Brown University is described, in which a computer program simulating a child learner acquires an interesting subset of English grammar from a "parent" program which knows the language.

Originator-supplied keywords included: - > front

Accession For	
NIC C&A	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By PER CALL JC	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	



Language Design, Computers and Brains *)

Henry Kučera

Brown University

In this paper, I will be concerned with investigating--in a preliminary way--the properties of natural languages (such as English, Russian, French, etc.) which make certain computer-based analyses, processing and learning of natural languages structures possible, as well as those language characteristics that are responsible for the formidable difficulties encountered in many of the more ambitious project of automatic language analysis. Although the question of computer-processing of natural languages is clearly of substantial practical importance (in such applications as word-processing, error detection in texts, various language-based expert systems and other products of Artificial Intelligence, or in machine translation), my primary concern will be to address the more fundamental issue of the differences between the two "information processors" involved in this comparative exercise: the biological one, the human brain, and the artificial one, the digital computer in its sequential processing mode as we know it today. In focusing on these issues, I hope to shed some light on the basic differences that human information processing (and thus the properties of human memory) exhibit in comparison with algorithmic automata. In conclusion, I hope to suggest some of the reasons why computers are superior to humans in performing some linguistic tasks but clearly substantially inferior in others.

The question of processing natural languages by computers is of intrinsic interest for yet another reason. If we assume--as we must--that languages have structure (more on that below), then that structure is specifiable by some set of principles generally referred to as "grammar" (in the broadest sense of the word). The types of formal grammars which generate (i.e. enumerate) all the well-formed strings of a language (whether it is an artificial or a natural one) have been extensively studied and their types determined. What

is of interest is the fact that the equivalence of formal grammars of different degrees of power and of automata with different properties has been established. The theory of formal grammars and the theory of automata are thus isomorphic in most important respects. The equivalence relations between the two are well-known: finite-state grammars correspond to finite automata, context-free phrase-structure grammars to pushdown store automata, context-sensitive phrase-structure grammars to linear-bounded automata, and unrestricted rewrite systems (such as certain types of transformation grammars) to Turing machines. Leaving aside, for the moment, the question of which type of grammar is adequate for a natural language (a matter of substantial controversy), the one-to-one correspondence between the hierarchy of grammars and automata should, at first glance, make the automata-based processing of languages a rather straightforward pursuit. The fact that this is definitely *not* the case tells us something about the nature of human language which formal grammars clearly do not capture, thus offering valuable insight into the organization of human information processing abilities.

The Mathematical Properties of Language

While we know next to nothing about the "origin" of human speech (even the dates offered by writers on this subject vary widely), historical linguistics offers us valuable evidence of the dynamism of language change; we do know that languages evolve and change over time and that language evolution thus must be connected with the mental resources available to humans in using the system for normal communication. There are two interesting properties which *all* human languages share and which point to the intuitive utilization of information-theoretic principles in the spontaneous construction of language systems by the members of any given community over time: the properties of redundancy and of efficiency. All natural languages exhibit both efficiency and redundancy, two contradictory characteristics which the linguistic systems balance against each other to achieve both a communicational usefulness and reliability.

Consider redundancy first: the basic "building blocks" of language are extremely limited--the distinct sounds which can be physiologically differentiated from each other for information signaling. The communicational power of language comes entirely from the concatenation of this small set of elements into an infinite set of expressions (words, sentences or discourses). But even as limited as the repertory of the basic sounds--the set of phonemes of a language--is (33 in English in the most common phonological analyses), only a small subset of their possible permutations can form actual words. An adult English speaker knows, for example, that *trip* is an English word. But he also knows that *tlip* is not an English word and he does not have to go to a dictionary to discover that fact; no English word can begin with *tl-*. But when faced with *trin*, the English speaker--although not recognizing the word--may have to take refuge in a dictionary. It is at least theoretically a possible English word because it does not violate any of the general constraints on permissible sequences of English sounds.

Even on this elementary phonological level, we thus find a substantial redundancy, the imposition of constraints on possible sequences and, consequently, the introduction of some information waste into the system, which is needed to enable us to communicate without overwhelming errors and misunderstandings. If every possible permutation of sounds were an actual English word, our communication system would be very efficient indeed and all our words could be very short; there would be no need for any word of more than four sounds: we could have over a million of those. But communicating in such a system would become extremely difficult. Our physiological limitations in producing and perceiving sounds and sound sequences, and the properties of human memory would make the learning and use of such a system practically impossible. But even if one could learn this distressingly efficient language, every noise, every imperfection or error that would destroy our perception of but a single sound, would disrupt our understanding, since the lack of redundancy of the system would not allow us to guess what it was that we may have missed. Worse still, if we heard one sound where another was intended, we would

have heard each time a legitimate, albeit unintended word. Thus, redundancy, a universal property of all languages, is one of our great communicational friends, just as it is a friend of the computer designer, who employs the same principle to detect potentially damaging transmission errors.

The constraints on permissible sequences of sounds that are utilized to achieve this redundancy may differ substantially from language to language: there are many languages in which a word can begin with an initial *tl-* cluster that English does not allow. While the principle of redundancy in language design is universal, its implementation is language specific. The same is true on higher levels; English--a configurational language that relies on word order to signal many grammatical functions--imposes severe restrictions on possible sequences of words within a sentence while languages with a "free" word-order, such as Latin or Russian, allow seemingly endless permutations of items. But to make this possible, these languages need to have an elaborate system of inflected forms and paradigms where a small set of endings is combined, in highly restricted ways, with the stems of the word to signal the syntactic relations that are achieved through word order in English.

Information theory provides a formal means of measuring the redundancy of a communication system. For natural languages, these measurements are complex and difficult but some overall estimates are possible. On the phonological level, we have calculated for several languages (English, German, Russian, and Czech) that--taking only the constraints on sound sequences within syllables into account--redundancy reaches about 50%. All languages, of course, also have restrictions on which syllables may follow each other: the overall redundancy estimate thus must be put at least at the 80% level. (For details, cf. Kučera and Monroe, 1968 and Kučera, 1975).

The redundancy phenomena on the phonological level present an important problem for the theory of first-language acquisition by young children. Children learn their first

language through exposure to the speech of their environment, without regard to any biological predispositions. Moreover, there is no systematic instruction involved in first-language learning. Yet the normal child quickly develops the general insights about admissible and non-admissible strings (possible and impossible words, as in the above example) which are *both* language specific (*not* universal and thus not "innate" in any reasonable sense) and based on access only to the "positive" data in the learning process, i.e. essentially to exposure to well-formed strings only. We have the proof (Gold, 1967) that the class of languages up to (and including) context-sensitive languages can be learned under certain assumptions. (Gold's definition of learnability is a technical one, known as "identification in the limit". A language L is said to be identified in the limit if, after some finite time, the learner "guesses" and continues to guess the actual grammar of L.) However, the requirements necessary for this learning to occur clearly violate developmental facts: so, for example, the learning can take place only if *all* the input data (i.e. the well-formed expressions "heard" by the child and used by him to choose the correct grammar eventually) are perfectly remembered throughout the learning process, obviously an unreasonable assumption for humans, especially when it comes to higher units, such as sentences. (The other possible approach is to view the input data as an infinite set of examples, surely not much of an improvement.) Moreover, Gold has also shown that even under these circumstances the learner would have to have access to *both* positive and negative evidence, i.e. be given information about well-formed expressions as well as ill-formed ones of the language; if only positive information is provided, then not even the class of finite-state languages is learnable. What is thus so interesting about the phonological redundancy example is the very fact that generalizations about a specific language (not universal generalizations or generalization physiologically determined) are actually acquired by all normal speakers of language *without* access to the negative evidence, i.e. to non-words. Phonology, too, of course, has its "grammar" - even if only a finite-state one. Consequently, the mathematical unlearnability proof without access to

negative evidence (i.e. ill-formed expressions) is contradicted by actual evidence, and a new language learning model, which is consonant with the empirical facts, is clearly needed.

Moving on to higher levels of language organization, particularly syntax, we can observe the relation between the restrictions on admissible combination of words to form grammatical sentences and the extension of the concept of redundancy to this communication level. The specification of the constraints that separates sentences from non-sentences is the grammar of the language on the syntactic level. Such a grammar is generally viewed as a quadruplet

$$G = \{V_t, V_n, S, R\}$$

where V_t is the terminal vocabulary of the language (lexical items), V_n the auxiliary vocabulary (equivalence classes of terminal symbols, such as nouns, verbs, noun phrases, etc.), S the privileged symbol of the units that the grammar is to enumerate (in our case "sentence") and R the set of rules that specifies the constraints on the well-formedness of S . We can then see that language can be defined as

$$L(G) = \{x \mid S \xrightarrow[G]{*} x \text{ and } x \in V_t^*\}$$

which can be read as language L , generated by the Grammar G is a set of all strings x such that every string x can be derived by the repeated application of the rules of the Grammar G and that all x 's are elements of the free semigroup V_t^* , i.e. consist of symbols of the terminal vocabulary only. Different types of grammars, from finite state to unrestricted ones, then differ only in the form of the rules which are permitted in the derivation. The restrictions of what are and what are not well-formed sentences, as specified in the grammar rules, then makes the assignment of structural description to a sentence (i.e. automatic parsing) possible. Without the constraints, the problem of ambiguity (cf. below) would make the task impossible.

The other side of the coin in language design is efficiency. Linguists have observed repeatedly that words which are used very frequently tend to be short. We even

abbreviate words as they become more common: *telephone* to *phone*, *airplane* to *plane*, *television* to *TV*--or *telly*, if one lives in England--and so on. Computer analysis of large samples of language texts now provides us with accurate data to support this general conclusion. In the one-million-word Corpus of Present-Day American English, also known as the Brown Corpus, compiled from samples taken from 500 different sources of 15 different genres and styles of writing, words accounting for 57% of the running texts (i.e. 57% of the one million word tokens) have four letters or less. But an entirely different situation comes to view if we construct a dictionary made from the Brown Corpus, i.e., a collection of different words (known in formal linguistics as "types"), with each word appearing only once in such a list. Here, words of four or fewer characters account for less than 9% of the dictionary. This discrepancy in itself suggests the communication efficiency of language: the system is so designed that it is the short words which are repeated often in an average text and thus accumulate high frequency figures. The longer words are used sparingly; the repeat rate of the truly long words is negligible. For every occurrence of a ten-letter word, there are eight occurrences of a three letter word, and for every instance of a twenty-letter word, there are 3,524 occurrence of a three-letter word. (For a description and various analyses of the Brown Corpus, cf. Kućera and Francis, 1967, and Francis and Kućera, 1982).

It is worthwhile to point out the similarity of this principle found in human languages to the design of artificial communication systems. In the International Morse code, the most frequent English letter, namely E, has the shortest symbol, one short signal, requiring minimum transmission time. The least frequent letters, J, Q, and Y, have the longest code, various sequences of three long signals and one short. What Samuel Morse did by planning, languages have achieved in their natural evolution. The same general principle is utilized in the design of variable-length character encoding, e.g. in the Huffman code system.

The efficiency which results in the much greater frequency of short words than long ones is further reflected in the highly skewed frequency distribution of the vocabulary in actual language use. Analysis of large samples of English shows, for examples, that 135 different words (mostly function words, such as articles, propositions, pronouns, but also some common content words, e.g. "man", "say") account for 50% of the tokens (running words) even in a very long text. At the other end of the frequency spectrum, about half of the words occur only once (these are known as *hapax legomena*). This striking frequency distribution was first expressed as a function by Zipf (1935) who attempted to develop an entire biological theory of language on the basis of this principle of "least effort." More accurate data for English of this striking phenomenon can be found in Kućera and Francis (1967) and in Kućera (1975).

The recognition of the word frequency distribution in natural languages and the identification of the vocabulary which accounts for large percentages of normal texts is, of course, of tremendous practical importance in any computer-based task where words need to be looked up in some list (such as dictionary). Even in relatively elementary word-processing aids such as spelling checkers, the speed of the checker's performance can be dramatically improved by loading the most frequent words into high-speed memory, thus minimizing access time in the dictionary look-up procedure.

Ambiguity

When examining those properties of natural languages which make their algorithmic processing difficult (and, on some levels, impossible), ambiguity emerges as the main culprit. There are at least two basic types of ambiguity that underlie the design of natural languages: formal ambiguity, manifested, for example, by the membership of the same lexical item in more than one equivalence class of non-terminals. The ambiguity of the sentence "Visiting relatives can be boring" is due to such a fact: the word "visiting" can be either a member of the class of adjectives (the sentence then meaning "The relatives who

visit can be boring") or of the class of gerunds (with the paraphrase "To visit relatives can be boring"). A phrase structure analysis of this sentence then yields two different parses, reflecting this structural ambiguity.

When attempting to design computer programs for sentence parsing, structural ambiguity is the source of substantial difficulties, especially in an analytic language such as English, where the suffixes are relatively poor predictors of the membership of a lexical item in a specific equivalence class. The word *round*, for example, can be an adjective, a noun, a verb, an adverb or a preposition. Humans, decoding English sentences, rarely notice any difficulties in such facts, selecting the proper class membership from syntactic environmental clues. A machine, which has no intuitive knowledge of such collocational clues, needs to be programmed to disambiguate all cases like these by applying appropriate rules which result, whenever possible, in a single equivalence class assignment to an ambiguous item. Only in this way can parsing proceed with any degree of success.

In our research, we have developed a technique of class assignments (known as "tagging") and a process of disambiguation of items with multiple tags which is statistically based. From our research, we have good estimates of the probabilities of a particular tag assignment to an item (so, for example, "spring" is much more likely to be a noun than a verb, but "make" is more likely to be verb than a noun), as well as--more importantly-- the statistics of transitional probabilities for pairs of tags. If one then faces, for instance, a series of six tags, each of them three-way ambiguous, the number of possible different sequences of tags for such a string is 729. Interestingly enough, using only local dependencies (i.e. transitional probabilities for pairs of tags), one can achieve a remarkable success in this disambiguation process, approaching 97% accuracy in the processing of an average English text. (For details, cf. Marshall, 1983). As the above example illustrates, the calculation may be complex and, should one encountered a long string of highly ambiguous items, could easily end up in a combinatorial explosion of

possible solutions. Fortunately, in practice, the strings of ambiguous items are quite limited even in an analytic language like English, so that the processing complexity is tolerable. Of particular cognitive interest is the locality principle exhibited in the process, where the dependencies between adjacent tags offer sufficient information for a high degree of accuracy of analysis.

Semantic ambiguity is essentially the consequence of another universal fact, the use of polysemy, which can be viewed as yet another manifestation of the efficiency principle in natural language design. Clearly, if every entity or concept, and every connotation of such entities and concepts were to be expressed by a different lexical item, we may indeed have a communication system of high precision but immense vocabulary, unmanageable within the constraints of human memory. In denotative and connotative partitioning of reality into discrete linguistic forms, all languages thus need to resort to a degree of polysemy, i.e. assign a single word more than one meaning. This kind of ambiguity, of course, was the major curse of the machine translation experiments of the 1950's and 60's and still is in those project which aim at mechanical translation. The story that the English sentence "His spirits were low" was translated into Russian as "He had almost no vodka left" is probably apocryphal but, nevertheless, revealing. The main problem illustrated here is not polysemy *per se* but rather the fact that different languages exhibit different principles in making use of this factor in language design (in this case, the English usage of "spirits" for both mental state and alcoholic substance, an ambiguity not shared by Russian.) Such language-specific strategies were systematically discussed by Whorf (1962) in the development of his theory of linguistic relativism. Even in the most basic vocabulary, such as the names of parts of the body, languages partition reality differently. In Russian, for example, the word "ruka" means *both* arm and hand, "noga" both leg and foot. Without an extensive examination of large amounts of context, the correct translation of "ruka" into English, which forces a decision between "arm" and "hand," is simply not possible.

Of greater interest still, however, is the systematic strategy in natural language design to use a hierarchical semantic organization principle to achieve communicational efficiency. This phenomenon, sometimes known in linguistics as the "theory of markedness," achieves signal saving by a principled use of polysemic features of one element as opposed to a related element which remains unambiguous as to that particular polysemic property. Let me summarize this interesting phenomenon briefly here with respect to lexical items (although it also can be applied to some grammatical categories). The definition and rationale of the theory can be found mainly in the writings of the members of the Prague Linguistic School, principally those of Roman Jakobson, who was the first to propose the extension of the markedness concept from phonology to the lexicon and to grammar. Jakobson's first formulation and justification of the theory was given in Jakobson, 1932; the essence of the argument was the rejection of the conventional notion that contrastive categories are "equal" members of the opposition relation (Jakobson's term is "gleichberechtigt"); instead Jakobson tried to show that the relations are hierarchical, with one of the categories, the unmarked one, characterized by the non-signaling of a property that the other, marked category, explicitly contains. Although Jakobson's basic concepts have not changed in any significant ways since first presented, I will cite here his most recent formulation (Jakobson, 1971):

"The general meaning of a marked category states the presence of a certain (whether positive or negative) property A; the general meaning of the corresponding unmarked category states nothing about the presence of A, and is used chiefly, but not exclusively, to indicate the absence of A. The unmarked term is always the negative of the marked term, but on the level of general meaning the opposition of the contradictories may be interpreted as 'statement of A' vs. 'no statement of A', whereas on the level of 'narrowed', nuclear meanings, we encounter the opposition 'statement of A' vs. 'statement of non-A'."

The American linguist, Joseph Greenberg, attempted to integrate the Praguian

markedness theory into his concept of implicational universals. The logical consequence of the definition of markedness is that the implication relation between the marked and unmarked members of the opposition is a unilateral one. In practical terms, the unmarked member can fulfill the function of the marked one but not the other way around. It is not accidental that the most convincing examples of markedness (which also makes claims as to its applicability to grammar) are lexical; Greenberg, for example, begins with *man* which is said to be the unmarked term having two functions, the general, denoting a human being (as in *Man is mortal*) and the other, denoting a male (as in *I saw your wife with a tall man last night*). Its opposite, *woman*, is then considered to be marked, stating the presence of the distinguishing feature, i.e. femininity, and thus having only a specific but no general function. Jakobson's initial example (1932), significantly enough, is also lexical: the Russian word *osel* "donkey" (considered unmarked) and the feminine *oslica* "female donkey" (marked). The term *osel* clearly has the same scope as *man* in Greenberg's example. Notice that the basic premise of the markedness theory is semantic. It is the semantic feature of the marked form that needs to be identified and it is the two meanings of the unmarked form (the general and the nuclear) which need to be examined.

Interesting, from the point of view of language design, is also the fact, which Greenberg tried to demonstrate, that the unmarked form is more frequent than the marked one in actual usage, something which one would indeed logically expect of an item which has a broader range of meaning. This correlation holds up well when we analyze marked and unmarked lexical items and their frequency. With regard to more general categories, however (such as the grammatical categories), the actual evidence is highly contradictory; some data support the correlation and others contradict it; in still other cases, such as tense forms, the issue turns out to be statistically undecidable. The details are complex but can be found in some of my previous articles (particularly in Kučera, 1980 and 1982).

To see the relation of markedness to general language design, consider first the fact that the markedness relation is essentially a special case of the relation of hyponymy. The term hyponymy is used by Lyons (1977:291 ff.), for example, as a more suitable designation for what, in logic, has been often discussed in terms of class-inclusion. The hyponymy relation can be best illustrated on examples involving the relation of simple lexical items: the word *rose* is a hyponym of *flower*, with the word *flower* being the superordinate term of the relation. If we consider the extension of the lexeme (in the logical sense of extension), then the superordinate term is more inclusive: *flower* includes not only *rose*, but *daffodil*, *tulip*, etc. In terms of intension (again in the logical sense of intension), the hyponym is more inclusive: *roses* have all the properties of *flowers* plus additional properties which distinguish them from *tulips*, *daffodils*, etc.

Hyponymy is definable by a unilateral implication. So, for example, the verb *waltz* can be established to be a hyponym of *dance* by the virtue of the implication: *She waltzed all night --> She danced all night* (but, of course, not the converse). This kind of definition of hyponymy by means of a unilateral implication also allows us to define synonymy as bilateral implication, or symmetrical hyponymy.

As Lyons also suggests, the Praguian markedness relation is, essentially, a special case of hyponymy. The principal difference is that the unmarked term has two meanings, the general (which gives it the usual status of a superordinate term) and the narrow or nuclear, which has a more specific sense, depending on context, and puts it in opposition to the marked term. Lyons suggests that the markedness relation may differ from the simple hyponymy relation by its potential of being reflexive: *Is that dog a dog or a bitch?* is meaningful, though rather odd. (Lyons, 1977:308).

In spite of a number of controversial issues that surrounds the markedness hypothesis, it does indeed offer one important insight, at least as far as the mapping of semantic concepts into language form is concerned: there is a systematic economy of

expression which uses ambiguity to achieve a more efficient use of the limited formal entities available, employing two terms for three distinct concepts and utilizing context to assign specific meanings.

Language Learning

In this section, I will focus on the description of a project, described in detail elsewhere (Shrier, 1977; Liberman, 1979; Kučera, 1981), which demonstrated that a computer can "learn" some aspects of a language which is a fairly large and interesting subset of English. I will deal here only with syntax, and will make no claims about the ability of machines to learn to construct meaningful discourse. In particular, I will show that the learning process results in a "creative" machine capacity in the sense that the computer can produce sentences which are not only syntactically well-formed but also new, i.e. had not occurred as part of the input into the learning program.

That the problem of first language acquisition is not only interesting but also highly important for our understanding of human mental functioning becomes clear when one considers the very basic facts that need to be taken into account in the explanation of the language acquisition process. First of all, all normal children learn the language of their community; there are no discernible racial or ethnic predispositions that play a role in this process. Thus a Chinese child transplanted into an English-speaking environment will learn English with complete native fluency and, conversely, a child of American parents, brought up in China, will speak the language of his Chinese community without any foreign "accent". Thus we clearly deal here with a learning process, with the resulting linguistic competence directly reflecting the learning environment. Moreover, first-language learning appears to occur almost spontaneously, without any explicit instructions. Mothers certainly do not normally tell their children which word is a noun and which a verb, not to speak of the syntactic rules governing the admissible combinations of these grammatical categories. And yet the child eventually acquires a

knowledge of the language which includes an awareness -- mostly unconscious -- of some regularity or structure of the language, something that we can call "grammar" in the broadest sense of the word. When linguists speak of the "creative" nature of linguistic competence, they essentially refer to the ability of the native speaker to understand sentences which he had never heard before and to produce "new" sentences of his own, albeit within the structural principles of the language, so that these "new" sentences are understood by other members of his language community. At first glance, we then seem to be witnessing in the process of first-language acquisition a classical case of induction of certain abstract principles -- of the grammar in the broad sense -- from "raw" input data, i.e. the speech that the child hears around him. While the input into this learning process -- the sentences heard by the child -- is obviously finite, the eventual competence of a mature speaker is potentially infinite in that he can produce an unbounded set of well-formed sentences of the language. Thus we have to assume the acquisition on the part of the learner of rules that are not only generizable to entire classes of items (such as common nouns, for example) but that also have at least some recursive properties. In short, we seem to be witnessing, in first-language acquisition, a truly remarkable process of inductive inference at work.

If it can be shown that there are significant aspects of the language acquisition process that can be successfully simulated by a computer program, this fact would surely introduce an important element into the discussion of language acquisition by humans. At the very least, an investigation of this kind should be able to give us some insight into the question of which specifically linguistic "innate" properties may be required in explaining the acquisition process and which aspects of this process may be explainable from much more *general* principles of intelligent behavior.

The language learning project, which we have been conducting at Brown University, is of particular interest because it is in principle consistent with the assumptions of neural

network learning and organization theories as discussed, for example, by Cooper (1984) and Anderson (1984). Our language learning algorithm operates with two "machines", a parent machine and a child machine. This terminology should not be interpreted as referring to any specific kind of hardware, however; it is nothing more than a convenient shorthand for the two main procedures in our computer program which assumes the two intuitively necessary elements in the language learning process, the "parent" who knows the language that is being transmitted to the new generation, and the "child" who is learning it. In our project, the parent machine is programmed to know all the relevant aspects of the language's grammar perfectly. The child machine, on the other hand, is --at the start of the learning process -- a *tabula rasa*, with the single exception of the terminal vocabulary, i.e. the lexical words of the language which the child machine is assumed to know. (This is equivalent to assuming that vocabulary can be learned by memorization.) It should be emphasized, however, that the child machine does *not* know the grammatical function of the vocabulary items, i.e., for example, which words are nouns, which verbs, etc. The proper partitioning of the vocabulary into grammatical classes is clearly essential in order for the learning program to succeed and is thus one of the tasks of the learning algorithm.

The learning process, embodied in our algorithms, involves essentially two types of interaction between the parent machine and the child machine:

(1) The parent machine produces a series of sentences which are well-formed in terms of its grammar. The child machine "listens" to these sentences. Note that the only input into the learning process is thus an unanalyzed surface sentence, what is sometimes loosely called the "surface structure" of the sentence. There is no information about *some* underlying "deep structure" nor any bracketing of the surface string. In this respect, our algorithm differs from some other mathematical models of language learning in which much more information is provided about the input data.

(2) The child machine attempts to "speak" (i.e. generate strings of words) by

modifying these sentences through the utilization of other vocabulary items at its disposal or through an interchange of vocabulary items within the sentence. In this speaking mode, the child machine produces some strings which are sentences as well as some which are non-sentences.

(3) The "speech" of the child machine is either accepted by the parent machine or rejected by it. This step thus serves as a binary reinforcement device. Notice that the parent machine thus has a dual function in our approach: it serves as a generator of well-formed sentences *and* as an acceptor of the strings produced by the child machine, accepting them as well-formed, or rejecting them as ill-formed. The child machine, in turn, utilizes both sets of information in its learning strategy that eventually results in the acquisition of the same functional competence as the parent machine has.

The abduction approach that we have used in our learning algorithms is thus not devoid of some initial assumptions. While we clearly do not posit a specialized and innate "faculty of language" with specific linguistic properties, we do make the *initial abductive* inference that there is regularity in language -- an inference that we assume the child, learning the language, to be able to make. It is precisely this assumption that, in our approach, motivates the initiation of the search for equivalence classes and syntactic rules. But the discovery of this regular structure from the "raw" data can be considered to be a function of a larger cognitive capacity of humans, applicable to other intellectual processes and learning experiences aside from language.

The fact that the various experiments with computer simulation of language acquisition have been promising does not mean, of course, that the difficult problem has been "solved". There are a number of controversial points which developmental psychologists and language-acquisition specialists still debate. The very role of some reinforcement (simulated in our programs by the binary approval/disapproval response of the parent machine), is controversial. Clearly, parents do not offer a simple binary

reinforcement to children acquiring their first language. I would like to argue, however, that any assumption that there is no behavioral indication which the learner could interpret as reinforcement is not plausible either. The very fact that mothers, for example, do repeat a child's simplified sentences in fuller form -- something that child language researchers have observed -- or, for that matter, the simple fact that a child may or may not be understood when speaking, would certainly seem to play a role in this complex learning process.

Conclusion

Natural languages and their processing by computers offer significant insights about the fundamental differences between the two information processors involved in all such endeavors, the human brain, which has fitted language design to its processing structures through evolution, and the algorithmic machines built by men.

As is the case with calculations, computers are greatly superior to humans whenever an exact language-based look-up or exact comparison is involved. If properly constructed, a spelling checker (technically known as a "spelling verifier" since it simply flags putative errors not found in a stored word-list), can proofread a 500 word document in as little as 8 seconds on an average personal computer. A human proofreader would need about 8 minutes to do this, i.e. 60 times as long. When the task becomes more ambitious, however, and the machine is programmed to actually correct the misspelling (i.e. to suggest correct spellings), the machine's performance deteriorates rapidly, both in quality of the suggested offerings and in speed. Many "correctors" now commercially available tend to present six to eight suggestions to the user for every misspelling and make take as long as 10 seconds per error to do so. Moreover, some of the corrections produced are quite unreasonable. A good human speller, on the other hand, can correct the misspelling in a fraction of this time, without considering a whole array of substitution candidates. The reason for the difference between simple verification and correction lies, of course, in the

difference in the processes involved: verification involves nothing more than *simple* matching of an item against a list, while correction involves the process of *associating* an ill-formed string with the best well-formed string, by identifying the linguistically salient features of the misspelling in order to find the correct substitute. Only those correctors (which are now becoming available) that attempt to simulate the process of human association achieve a better speed and greater precision characteristics, approaching human performance.

A similar phenomenon can be observed with respect to other aspect of language processing. Speech synthesis by computers is, essentially, a solved problem; it requires basically the translation of a complex algorithm of predetermined acoustic features, representing some "standard" form of speech, into the appropriate signals. Automatic speech analysis, especially in the case of continuous speech, is much more difficult. The problem of identifying constituent boundaries (between words and phrases) as well as acoustic invariants in the input of different speakers involves the extraction of the salient features that make it possible to map diverse signals into a single representation.

In the realm of grammar, the tasks show a familiar pattern as well: People--be it children or students of a foreign language--understand before they speak. Students of a foreign language can normally read the language before they can speak it. The opposite is the case with computers. It is considerably easier to program computers with reasonably adequate grammars to generate grammatical (although not necessarily semantically congruous) sentences. It is much more difficult to develop computer parsers that assign a structural description (such as labeled bracketing) to actual language input. Sentence production by machines is thus considerably easier than their parsing.

All these revealing contrasts in natural-language processing reinforce the well-known differences between brains and computers: Brains are parallel processors, particularly adapt at pattern recognition and similarity detection, utilizing various disambiguation

strategies and "fuzzy" logic in achieving their processing goals. Computers are sequential processors, requiring identity in search and comparison, and a yes/no answer to every question. Approximations, associations and disambiguations are essentially alien to computers; if these processes are required, they must be simulated, often at a great processing cost.

Technology, of course, is evolving precisely in the direction which is likely to reduce the differences between the two types of processors. There are already parallel processors in the design stage where a great number of simultaneous processing units is wired to work on a problem simultaneously. When these automata become powerful enough, natural language processing by machine will enter a new era and our understanding of the properties of language design--as determined by our brain functions and capabilities--is bound to increase immeasurably.

*) The research for this report was supported, in part, by Grant No. N0001481K**0136** from the Office of Naval Research.

REFERENCES

- Anderson, J. A., 1984. "Neural Models and Little about Language," *Biological Bases of Language*, ed. by D. Caplan, A. Roche-Lecourgs, and A. Smith (MIT Press, Cambridge, Mass.).
- Cooper, L. N., 1984. "Neuron Learning to Network Organization," *J. C. Maxwell Sesquicentennial Symposium* (Noth Holland, Amsterdam).
- Francis, W. N. and H. Kućera, 1982. *Frequency Analysis of English Usage: Lexicon and Grammar* (Houghton Mifflin Co., Boston).
- Gold, M. E., 1967. "Language Identification in the Limit," *Information and Control*, 10, 447-474.
- Greenberg, J. H., 1966. "Language Universals," in *Current Trends in Linguistics*, III, T. A. Sebeok, ed. (The Hague), 61-112.
- Jakobson, R., 1932. "Zur Struktur des russischen Verbums," in *Selected Writings*, 2 (The Hague), 3-15.
- Kučera, H., 1975. *Computers in Linguistics and in Literary Studies* (Brown University, Providence, R.I.).
- Kučera, H., 1981. "The Learning of Grammar," *Perspectives in Computing*, Vol. 1, No. 2, 28-35.
- Kučera, H. and W. N. Francis, 1967. *Computational Analysis of Present-Day American English* (Brown University Press, Providence).
- Kučera, H., 1982. "Markedness and Frequency," in *COLING 82*, J. Horecký, ed. (Amsterdam), 167-173.
- Kučera, H. and G. K. Monroe, 1968. *A Comparative Quantitative Phonology of Russian, Czech, and German* (American Elsevier, New York).
- Liberman, F. Z., 1979. *Learning by Neural Nets* (Doctoral dissertation, Brown University, Providence, R.I.).
- Lyons, J., 1977. *Semantics 1* (Cambridge).
- Marshall, I., 1983. "Choice of Grammatical Word-Class Without Global Syntactic Analysis," *Computers in the Humanities*, Vol. 17, 139-150.
- Shrier, S., 1977. *Abduction Algorithm for Grammar Discovery*, Department of the Navy Technical Report (Division of Applied Mathematics, Brown University, Providence, R.I.).
- Whorf, B. L., 1962. *Language, Thought and Reality, Selected Writings*, (Cambridge, Mass.).
- Zipf, G. K., 1935. *The Psycho-Biology of Language* (Houghton Mifflin Co., Boston).